

סיבוכיות

© ארזים

23 במאי 2017

1 אי-דטרמיניזם

1.1 המחלקה NP

ניזכר שראינו את השפות CSAT, CVAL בשיעורים קודמים

תרגיל אם ניתן לפתור בעילות את בעיית ההכרעה CSAT, אז ניתן לפתור בעילת גם את בעיית החיפוש המתאימה.

יש דוגמאות נוספות למקרה הזה - מעגל המילטון, קליקה בגודל k , משפטים מתמטיים יכחים.

הגדרה 1.1 לבעיית הכרעה $S \subseteq \{0, 1\}^*$ יש מוודא יעיל אם יש פולינום $t : \mathbb{N} \rightarrow \mathbb{R}$ ואלגוריתם/מכונת טיורינג V כך שמתקיים:

1. לכל $x \in S$ יש y עם $|y| \leq t(|x|)$ וכן $V(x, y) = 1$.

2. לכל $x \notin S$ ולכל y מתקיים $V(x, y) = 0$.

וכן V רץ בזמן פולינומי באורך הקלט שלו.

יהי $R \subseteq \{0, 1\}^*$ המוגדר על ידי

$$R = \{(x, y) \mid V(x, y) = 1, |y| \leq t(|x|)\}$$

זוהי בעיה במחלקה P, והמכונה שמכריעה אותה היא V .

הגדרה 1.2 NP היא מחלקה כל בעיות ההכרעה שעבורן יש מוודא יעיל שכזה.

נגדיר כעת את NP בדרך אחרת - בעזרת אי דטרמיניזם.

הגדרה 1.3 מכונת טיורינג לא דטרמיניסטית היא מכונת טיורינג בה פונקציית המעברים היא מהצורה

$$\delta : \Gamma \times Q \rightarrow P(\Gamma \times Q \times \{\leftarrow, \rightarrow\})$$

מכונה שכזו מקבלת קלט x אם יש מסלול חישוב מהקונפיגורציה ההתחלתית לקונפיגורציה מקבלת. מכונה כזו מכריעה קלט x בזמן t אם כל מסלול חישוב של המכונה על x מסתיים אחרי t צעדים.

במילים אחרות, מכונת טיורינג לא דטרמיניסטית בוחרת בכל צעד חישוב מבין כמה אפשרויות. אם יש סדרת בחירות שמובילה לקבל של x , אזי המכונה מקבלת את x .

הגדרה 1.4 נגדיר $\text{NTIME}(t(n))$ להיות אוסף כל השפות המוכרעות על ידי מכונה אי דטרמיניסטית שרצה בזמן לכל היותר $O(t(|x|))$ על קלט x .

הגדרה 1.5 נגדיר

$$\text{NP} = \bigcup_c \text{NTIME}(n^c)$$

טענה 1.6 שתי ההגדרות של NP שקולות.

הוכחה: נניח שקיים מוודא יעיל, ונראה מכונת טיורינג לא דטרמיניסטית. המכונה תכתוב באופן לא דטרמיניסטי מחרוזת אפשרית y באורך לכל היותר $t(|x|)$, ואז תריץ את המוודא היעיל על (x, y) כעת נניח כי יש מכונת טיורינג לא דטרמיניסטית, ונראה שקיים מוודא יעיל. העד y יהיה סדרת הבחירות של המכונה באורך לכל היותר $t(|x|)$. המוודא יוודא שאכן y מתאר מסלול חישוב חוקי ומקבל של x . ■

משפט 1.7 (קוק-לווין) CSAT שלמה למחלקה NP תחת רדוקציות \leq_L . גם 3SAT כזו.

הוכחה: ראינו כבר כי $\text{CSAT} \in \text{NP}$ (למשל לפי ההגדרה עם מוודא יעיל). תהי כעת A שפה מתוך NP עבודה יש מכונת טיורינג לא דטרמיניסטית שרצה בזמן לכל היותר $t(n)$ שמכריעה את A . נריץ את הרדוקציה אל CVAL כפי שראינו בשיעור שעבר, רק שהפעם בנוסף לקלט x נדמיים גם שיש t משתנים y . לא קשה לוודא שמקבלים קלט חוקי עבור CSAT כנדרש, כאשר משתני המעגל הם y . הרדוקציה תשתמש במוודא V (כלומר בפונקציית המעברים שלו). ברור שהרדוקציות אכן ביזרון לוגריתמי, ולכן סיימנו. ■

הגדרה 1.8 SAT היא אוסף כל הנוסחאות הספיקות בצורת 3SAT.CNF. היא אוסף כל הנוסחאות הספיקות בצורת 3CNF.

משפט 1.9 שתי השפות שהגדרנו עכשיו הן NP שלמות תחת רדוקציות \leq_L .

הוכחה: (סקיצה) ברור כי שתי השפות נמצאות בתוך NP. נראה למשל $\text{SAT} \leq_L \text{CSAT}$. כלומר, בהינתן מעגל בוליארי, צריך להוציא נוסחת CNF שתהיה ספיקה אם ורק אם המעגל ספיק. נניח שמשתני המעגל הם x_1, \dots, x_n ושהמעגל בגודל S . נגדיר S משתנים חדשים, z_1, \dots, z_n . המשתנה z_i יתאים לשער שמספרו i . השער מסומן על ידי \wedge (באופן דומה עבור \vee, \neg) ובניו הם השערים j, k , נכתוב פסוקיות CNF המסתפקות אם ורק אם $z_i = z_j \wedge z_k$. אם השער i הוא שער קלט, למשל x_j , נכתוב פסוקית שמשמעה $z_i = x_j$. הנוסחה הסופית שנוציא תהיה \wedge של כל הנוסחאות הללו, וכמו כן נוסיף את המשתנה z_{output} שמתאים לשער הפלט.

■ לא קשה לראות שזו רדוקציה חוקית בזמן לוגריתמי בין השפות (אפילו אל 3SAT).

מספר בעיות NP שלמות תחת רדוקציות \leq_L :

1. $CLIQUE_k$ - בהינתן גרף ומספר k , האם יש בגרף קליקה בגודל k לפחות.
 2. $HAMCYCLE$ - בהינתן גרף, האם יש בו מעגל המילטון.
 3. $COLOUR - 3$ - בהינתן גרף, האם הוא שלוש-צביע (בקודקודים).
 4. IS - קבוצה חופשית של קודקודים בגרף.
 5. $SUBSET - SUM$ - מקבלים k ועוד n אי שליליים בני n ספרות, וצריך להריע האם יש תת קבוצה שלהם שסכומה שווה k .
 6. $Integer Programming$ - בעיות תכנות לינארי בשלמים.
- דוגמה $SAT - 2 - MAX$** היא שפת כל הנסוחאות בצורת $2CNF$ ומספרים t כך שיש השמה שמספקת לפחות t פסוקיות של הנוסחה.

הערה 1.10 השפה $2SAT$, של כל הנסוחאות בצורת $2CNF$ הספיקות, שייכת למחלקה P .

משפט 1.11 $SAT - 2 - MAX$ היא NP שלמה תחת רדוקציות \leq_L .

הוכחה: נראה רדוקציה משפת $3SAT$. בלי הגבלת הכלליות, נניח שבכל פסוקית יש 3 ליטרלים שונים. בהינתן פסוקית $x \vee y \vee z$, ניצור ממנה 10 פסוקיות עבור נוסחת $2CNF$ שאנחנו עומדים ליצור. לשם כך נשתמש במשתנה חדש w הייחודי לפסוקית זו. הפסוקיות שנגדיר הן

$$x, y, z, w, \neg x \vee \neg y, \neg x \vee \neg z, \neg y \vee \neg z, x \vee \neg w, y \vee \neg w, z \vee \neg w$$

אנחנו נטען את שלושת הדברים הבאים:

1. אם השמה מספקת את $x \vee y \vee z$ אז יש דרך לקבוע את w עבורה יסתפקו בדיוק 7 פסוקיות.
2. לא ניתן לספק יותר מאשר 7 פסוקיות.
3. אם השמה לא מספקת את $x \vee y \vee z$ אז היא מספקת לכל היותר 6 פסוקיות.

הדרך להוכיח את אלה היא הצבת כל הערכים האפשריים וספירה. נקבל נוסחת $2CNF$, שהיא \wedge של כל הפסוקיות שהגדרנו. כמו כן, הפרמטר t יהיה 7 כפול מספר הפסוקיות בנוסחה המקורית. לא קשה לראות (לפי הטענה) שזו רדוקציה עם התכונות הנדרשות. ■

הגדרה 1.12 מכונת טירוינג אי דטרמיניסטית מוגבלת זיכרון היא מכונת טירוינג מוגבלת זיכרון עם פונקציית מעברים מהצורה

$$\delta : \Gamma \times Q \rightarrow P(\Gamma \times Q \times \{\leftarrow, \rightarrow\} \times \{\leftarrow, \rightarrow\})$$

כאשר הכיוון הראשון הוא בסרט העבודה והשני בסרט הקלט. מכונה כזו מקבלת קלט x אם יש סדרת צעדים מהקונפיגורציה ההתחלתית לקונפיגורציה מקבלת. סיבוכיות הזיכרון על קלט x היא מספר תאי הזכרון המקסימלי שהמכונה משתמשת בהם. נאמר כי שפה A שייכת למחלקה $NSPACE(s(n))$ אם יש מכונת טירוינג לא דטרמיניסטית מוגבלת זיכרון המקלט את A ועל קלט x משתמשת בזכרון $O(s(|x|))$.

הגדרה 1.13 מגדירים

$$\text{NL} = \text{NSPACE}(\log n)$$

$$\text{NPSpace} = \bigcup_c \text{NSpace}(n^c)$$

משפט 1.14 השפה STCON היא NL שלמה תחת רדוקציות \leq_L .

הוכחה: ראשית נראה $\text{STCON} \in \text{NL}$. הקלט שלנו הוא גרף על n קודקודים ושני קודקודים. נחזיק מונה שיספור מאחד עד n . בכל שלב ננחש קודקוד באופן אי דטרמיניסטי (כלומר ננחש מספר של קודקוד). נודוא שהוא מחובר לקודם, ונקבל אם הצלחנו להגיע אל t (התחלנו מהקודקוד s) לפני שהמונה הגיע אל הערך $n + 1$. הנכונות ברורה, וכן הסיבוכיות. נעבור לשלמות. ניזכר כי למכונה עם $O(\log n)$ זיכרון יש $n^{O(1)}$ קונפיגורציות אפשריות. בהינתן שפה $A \in \text{NL}$, תהי M מכונת טירוינג לא דטרמיניסטית שמכריעה אותה עם $O(\log n)$ זיכרון. נניח כי יש לה n^c קונפיגורציות אפשריות. בהנתן קלט x של M , נוציא גרף על n^c קודקודים, כאשר s הוא הקודקוד המתאר את הקונפיגורציה ההתחלתית, t את הסופית (בלי הגבלת הכלליות היא יחידה). לפי תיאור M נקבע האם בין שני קודקודים יש קשת. ברור שזו רדוקציה טובה. ■

ראינו בעבר כי $\text{STCON} \in \text{DSPACE}(\log^2 n)$. מכאן נקבל מסקנה:

מסקנה 1.15 $\text{NL} \subseteq \text{DSPACE}(\log^2 n)$

ממסקנה זו נובע משפט כללי יותר:

משפט 1.16 (סאביץ') אם $\log n \leq s(n)$ אזי $\text{NSpace}(s(n)) \subseteq \text{DSPACE}(s(n)^2)$.

נקבל את תמונת העולם הבאה:

$$\text{L} \subseteq \underbrace{\text{NL}}_{\text{STCON}} \subseteq \underbrace{\text{P}}_{\text{CVAL}} \subseteq \underbrace{\text{NP}}_{\text{3SAT}} \subseteq \underbrace{\text{PSPACE}}_{\text{TQBF}} = \text{NSpace}$$

$$\text{NL} \subseteq \text{DSPACE}(\log^2 n)$$

הגדרה 1.17 עבור שפה $A \subseteq \{0, 1\}^*$ נגדיר $\bar{A} = \{0, 1\}^* \setminus A$. זוהי השפה המשלימה.

הבחנות

$$\overline{\bar{A}} = A$$

$$A \subseteq B \iff \bar{B} \subseteq \bar{A}$$

הגדרה 1.18 עבור מחלקה \mathcal{C} של שפות נגדיר

$$\text{co-}\mathcal{C} = \{\bar{A} \mid A \in \mathcal{C}\}$$

דוגמה $\overline{\text{SAT}} \in \text{co-NP}$ היא שפת הנסוחאות בצורת CNF שאינן ספיקות.

טענה 1.19 $\text{co-P} = \text{P}$.

■ **הוכחה:** לא קשה לראות כי $A \in P$ אם ורק אם $\overline{A} \in P$ (מחזירים הפוך).

טענה 1.20 תהיינה C_1, C_2 שתי מחלקות עם $C_1 \subseteq C_2$. אזי

$$\text{co-}C_1 \subseteq \text{co-}C_2$$

הוכחה:

$$\overline{A} \in \text{co-}C_1 \iff A \in C_1 \Rightarrow A \in C_2 \iff \overline{A} \in \text{co-}C_2$$

■

מסקנה 1.21 אם $C \subseteq \text{co-C}$ אזי $C = \text{co-C}$.

■ **הוכחה:** ברור כי $\text{co-co-C} = C$, ולכן מהטענה נובעת ההכלה השנייה.

טענה 1.22 אם שפה A שלמה למחלקה C תחת רדוקציות \leq_L , אזי \overline{A} שלמה למחלקה co-C תחת רדוקציות \leq_L .

■ **הוכחה:** מבצעים את אותה רדוקציה בדיוק - אם $B \in C$, φ הרדוקציה בזכרון לוגריתמי אל A , אזי φ היא גם רדוקציה בזיכרון לוגריתמי של \overline{B} אל \overline{A} .

מסקנה 1.23 $\overline{3\text{SAT}}$ שלמה למחלקה co-NP תחת רדוקציות \leq_L .

היוריסטית, NP היא מחלקת השפות שקל לוודא עבורן הוכחה, co-NP היא שפת המחלקות שאין להן הוכחה. כדי להראות הכלה באחד הכיוונים, צריך להבין, למשל, איך מראים שלנוסחת CNF אין השמה מספקת.